CONNEXIONS™

The Interoperability Report

February 1989 Volume 3, No. 2

ConneXions—
The Interoperability Report tracks current and emerging standards and technologies within the computer and communications industry.

In this issue:

ConneXions is published by Advanced Computing Environments, 480 San Antonio Road, Suite 100, Mountain View, CA 94040, USA. Phone: 415-941-3399.

ISSN 0894-5926

From the Editor

This month Phil Almquist of Stanford University takes us behind the scene at INTEROP™ 88 and describes how the Show and Tel-net was constructed. Although the network operated successfully during the show, it was not a trivial task to organize, and many important lessons were learned.

Electronic mail is by far the most popular application in all the major computer networks. By the use of electronic mail gateways, the various networks can be linked together to form a worldwide *metanetwork*. This metanetwork is the subject of a new book by John Quarterman of Texas Internet Consulting. The book is called *The Matrix: Computer Networks and Conferencing Systems Worldwide* and will be published this year by Digital Press. Starting on page 10 John gives an overview of electronic mail and discusses some of the problems facing a user of one network when he tries to send a message to someone on a different network.

The OSI Protocol Suite is slowly becoming a reality through the efforts of the standardization bodies and associated organizations. As crucial as this standardization work is, we should not forget that early implementation and experimentation plays an equally important role on the "Path to OSI." The US government has already declared its intention to adopt OSI for all government networks when the protocols reach a mature state. The Navy has been performing some early experiments with OSI protocols running on top of the Defense Data Network (DDN). Robert Slaski of NetWorks One and Robert Cooney of the Navy describe this project in an article on page 16.

A reminder about our tutorials in April and June appears on page 9 together with a brief description of a new book on the X Windows System.

Oops! We recently discovered a printing error in the October 1987 (Volume 1, No. 6) issue of *ConneXions*. This issue was reprinted in the summer of 1988 and the reprints are missing a diagram of the Domain Name System on page 3. If you were a subscriber prior to October 1987 you would have received the original (correct) version, but if you ordered this issue at a later date you probably got the one without the diagram. If you'd like a replacement copy of this issue, please contact us at 415-941-3399.

The INTEROP™ 88 Network—behind the scenes

by Phil Almquist, Stanford University

Introduction

One of the goals of the organizers of INTEROP 88 was to include a fairly ambitious demonstration of TCP/IP interoperability. An internet was built on the show floor which consisted of a subnetted class B network, a class C network, and approximately 250 hosts and gateways representing 49 different vendors. A couple of the more unusual features of this network were that it included a wide variety of transmission media and featured a level of high speed connectivity to the DARPA/NSF Internet and corporate networks that would put even most major research university networks to shame. A separate Ethernet was used for the CMIP over TCP/IP ("CMOT") demonstration put on by the IETF "Netman" Working Group.

Purpose

This article describes the choices we made in the design of the network and some of the operational problems we encountered. There are several things this article is trying to accomplish. One is to satisfy the curiosity of those who saw the network and want to know what was "under the hood." A second purpose is to give network designers the benefit of our experience. A third purpose is to stimulate thought about how to make TCP/IP networks more "plug and go." A final purpose is to publicly recognize some of the people who made it all work. An earlier version of this article was given as a talk at the October 1988 IETF Meeting.

Multiple media

One of the things that we wanted to show was that TCP/IP is not tied to any particular physical medium. We therefore designed the network to include as many kinds of media as we could, even though there was some risk in doing so because we had no prior experience with some of them. The network we built included Ethernet (both thick and thin), Ethernet over twisted pair, Ethernet over fiber optic lines, PRONet-80, IBM/802.5 Token Ring, serial lines, and amateur packet radio. Several vendors connected other kinds of media to our network, including Hyperchannel, PRONet-10, T-1, and Ethernet over broadband.

Cables

In order to show off this diversity of media, we suspended the cabling from the ceiling, where it was in plain view. Unfortunately, the diversity wasn't as apparent as we would have liked, since most of the kinds of media use cables that are thin and black and therefore not readily distinguishable from one another. A number of people complained to me that the transceivers, which were hung on loops in the cable about a dozen feet above the floor, looked "messy." While this may be true, the reasons why this was done were purely pragmatic: transceivers had to be low enough to be reached with ladders in case problems developed during the show. (Once all the booths were constructed there would be no room to allow "cherry pickers" in the exhibit hall). We also didn't want to run the loops all the way down to the floor level where they could be hit by a passing forklift during the construction phase.

The biggest problem that we encountered with the cable plant installation was that the person who computed how much cabling would be needed was told that the ceiling was lower than it actually was. The resulting cable shortage caused nearly a day's worth of slippage in the installation schedule, a delay that cost us dearly.

Additional delays resulted when, several times during the course of the installation, the network was partitioned because people working inside the wiring closet in the network operations center (NOC) accidentally knocked cables loose. While those people probably should have been more careful, they were under significant time-pressure and also often somewhat sleep-deprived. We probably should have done a better job of designing the wiring closet.

Once the cable plant was installed, it generally worked quite well for us. Despite the fact that we had nearly two miles of hastily-installed cabling, only one segment had any kind of a problem during the show (surprisingly, that segment consisted of thick Ethernet which is generally quite reliable if properly installed). That problem vanished as mysteriously as it had appeared.

Topology

The network topology was chosen based on an expectation that there would be problems, both with hardware and with protocol implementations. We were particularly worried about broadcast storms. Thus, the topology was based on three principles. First, the network design should maximize the network's ability to function in spite of problems, especially problems in host software. Second, the network design should make it very hard for any failure to disable a substantial portion of the network. Third, the design should facilitate rapid detection and diagnosis of problems. Many traditional factors in network design, such as cost, long term maintainability, and topological constraints based on geography, could be pretty much ignored—although other circumstances such as free-hanging cable, continual pedestrian traffic at cable extremities, etc. dictated some pretty creative cable anchoring and protection techniques.

These principles led us to a design based on a backbone network composed of a large number of small subnets connected by IP routers. The subnets were connected into what was logically a tree structure. (See Figure 1). Tree topologies have the disadvantage that there is no redundancy, so any failure will partition the network. Failures near the root of the tree have particularly serious consequences. However, we felt that this disadvantage was outweighed by the fact that tree topologies make failure detection and fault isolation particularly easy. In order to minimize the disadvantage, we placed the root of the tree, including the backbone routers, into the network operations center where they were easy for us to access and monitor. Furthermore, we chose as backbone routers only routers with which we and the Internet community as a whole have had extensive experience.

To further facilitate rapid problem diagnosis, we made the subnets themselves very simple, with a minimum of active components. We didn't use bridges or repeaters. We did not mix media types within a single subnet (with one exception, a subnet which contained both fiber optic and twisted pair Ethernet).

I think that if I were doing this again I would make the same topological choices. As I will explain later, the large number of subnets did turn out to be a bit painful because of some last minute changes in the way we did routing.

continued on next page

The INTEROP 88 Network *(continued)*

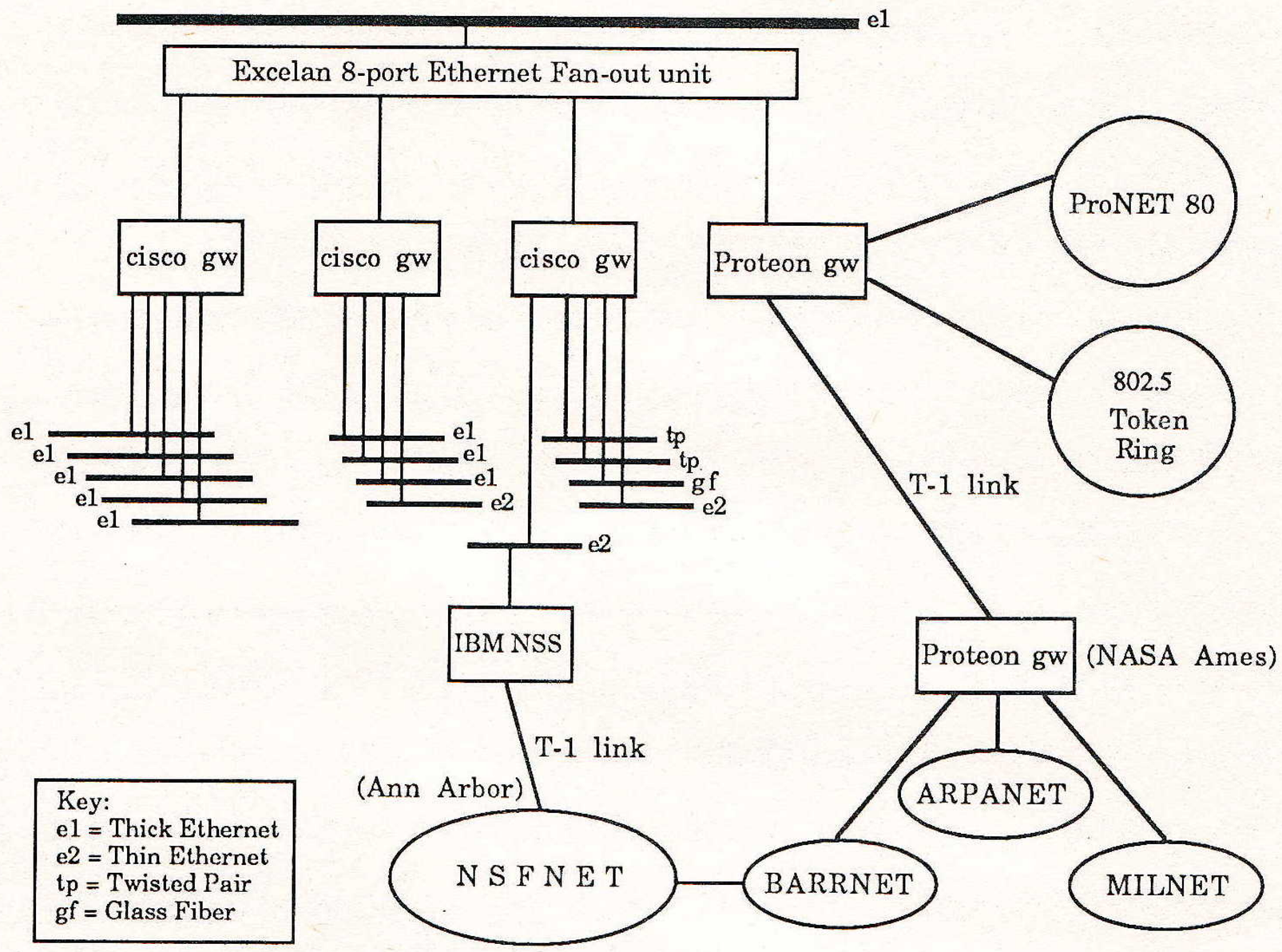

Figure 1: INTEROP 88 simplified network topology

Domains

When it came to the Domain Name System and host tables, what we did turned out to be far less effective than we expected. We requested a domain (ShowNet.COM) from the NIC for use at the show. We then sent to all the participating vendors questionnaires which they were supposed to fill out describing the hosts they planned to attach. When we got the questionnaires back, we used the information they contained to assign IP addresses and to create the master zone file for the ShowNet.COM domain. I wrote a program that read that master zone file and created NIC-format HOSTS.TXT file and the master files for the related IN-ADDR.ARPA zones.

Unfortunately, most of the questionnaires were not returned until after we were busy building the network and required personal followup with the vendors because they were filled out incorrectly. Since we had expected that this would be pretty much done before we started building the network, we hadn't arranged to have anybody at the show whose job it would be to handle the questionnaires, nor did we have any way to access and edit the zone file until the Internet connections came up. The problems were compounded by the fact that I, whose job description said I probably should have been dealing with the questionnaires, was instead working on building the network since the construction was way behind schedule.

Name servers

We had arranged to have three authoritative domain servers (two off-site) for the ShowNet.COM domain. Because this was a temporary network, we found it pragmatic to violate a couple of rules of good internet practice: we assigned very short (30 minute) TTLs to our domain data and used zone transfers to update the off-site domain servers. The domain server at the show also provided, via anonymous FTP, a NIC-format host table for those systems that do not yet support the domain system.

With this also, things did not go smoothly. The first problem was that, due to some miscommunication with the NIC, the domain application was initially rejected and did not get approved until a few days before the show. By that time, we had missed the NIC's weekly database update cycle. Fortunately, they were willing to do a special update for us. Our collective sigh of relief was premature, however, because their procedure which updates the root name servers chose that moment to break, so many of these servers were not informed about our domain. Luckily, Paul Mockapetris happened to walk in while we were considering how to solve this problem, and was able to update a number of the root servers by hand.

Even once the root server situation got straightened out, we still had some problems locally. The on-site domain server was installed late, due to the general schedule slippage. It also turned out to be non-trivial to get the domain server running correctly, partly because there were some typos in the master file and partly because nobody there knew much about the VMS version of BIND. We also discovered, much to our surprise, that the `/etc/hosts` format has become so ubiquitous that at least one vendor who needed a host table wasn't sure how to deal with the NIC format!

Network management

Network management was another casualty of the schedule slippage. We had an excellent set of tools available to us, including a protocol analyzer, a "smart" Ethernet terminator, and a workstation running Wollongong's version of the NYSERNet SNMP tools. Unfortunately, nobody had time to train our NOC personnel in how to use them. There were also other schedule-related problems: the patch panel that was intended to make it easy to attach the protocol analyzer to any desired subnet was never installed, and nobody had time to generate correct configuration files for the SNMP tools.

Routing

The routing problems we had to face were perhaps the most interesting part of the network design. We thought that routing between the various subnets of the show was going to be quite simple. We chose to use the *Routing Information Protocol* (RIP) because it is the lowest common denominator protocol. Our topology gave us control of all of the routers in the backbone network.

However, when we were setting up the show network, some vendors expressed concern over this plan. They were worried because some older 4.2BSD-based hosts are so broken that they participate in the RIP protocol even when they are not gateways. If such a host were misconfigured at any time during the show a black hole could easily result. Because of these concerns, we configured the routers in the backbone to believe routing updates only from each other. Static routes were used to access subnets that vendors spawned in their own booths. Although this system did prevent black holes, maintaining the static routes was painful.

Two T-1 paths

Routing to the corporate networks that were attached was also handled using static routes. However, routing to the Internet was a much more complex problem. We had available to us a direct connection to the NSFNET, but we also had a T-1-speed connection to NASA's Ames Research site. Since either path provided us good connectivity to the Internet, we needed to come up with an "appropriate" way to divide external traffic between the two.

continued on next page

The INTEROP 88 Network *(continued)*

We considered several factors when deciding how this ought to be done. The one that weighed most on our minds was that one or both of the links might prove unreliable. NSFNET, although running quite well, was still very new. Both connections depended on new T-1 circuits, and my previous experience with new T-1 links had not exactly been positive.

An additional concern was that the NSFNET link was being provided and operated by IBM. This would place us, as network operators, in a very unpleasant position if problems with the NSFNET connection started causing failures in other vendors' demos. Finally, we knew that we wanted to make use of both paths. The NSFNET path was the best path to most networks connected to NSFNET regional networks, while the NASA path was a better way to get to most networks hanging off of Arpanet or MILNET.

After considering these various factors, we concluded that we needed a scheme that would support automatic rollover to the other path if one of the paths failed. We also wanted to reserve for the network operations center the ability to manually force routes to use one of the paths.

The solution

I therefore devised the following scheme: One of the routers in the network operations center would be responsible for EGP exchanges with the NSFNET. This router could, under administrative control, be told which if any of the EGP routes it learned from NSFNET it should believe. Those it believed it advertised via RIP into the show network; packets destined for other Internet locations followed a default route out the NASA path. Similarly, we could control paths taken by return traffic in a very gross way by turning off announcement of the show net on one of the paths.

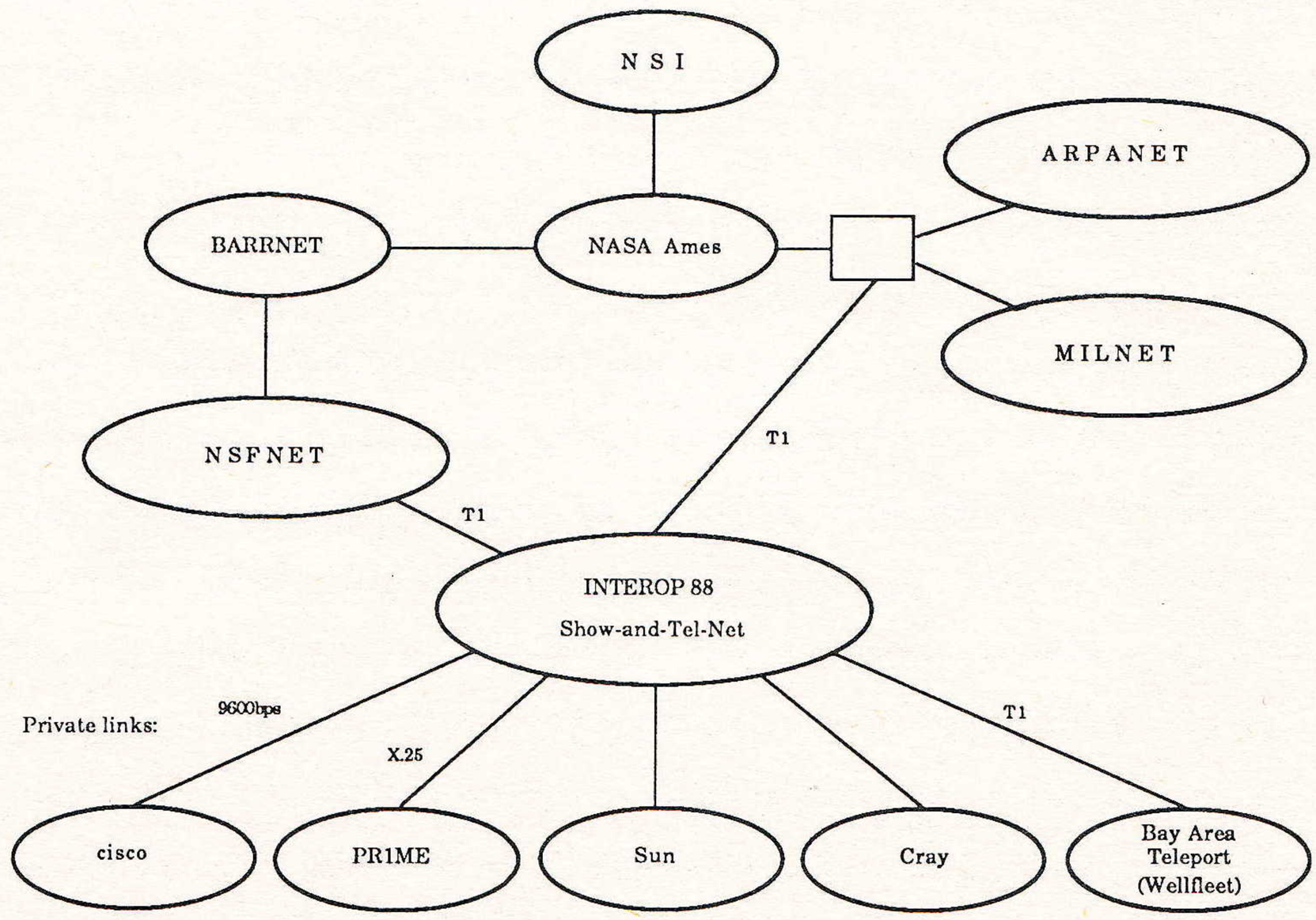

Figure 2: INTEROP 88 logical map

Although this scheme would work well for us as network managers, we realized that it had one important drawback: IBM in its demos might want to use NSFNET to get to some location that the network manager had decided should be routed through NASA. To avoid potential conflict, we needed to provide a way for IBM to operate their own external routing policy within their booth. To do this, we asked IBM to provide a machine which would EGP peer with the NSFNET and distribute appropriate RIP routes to other machines in their booth. We placed the IBM booth on its own Ethernet segment, so that their RIP traffic would not get intermixed with ours. We also gave their segment its own IP network number, so that EGP would treat that segment as separate from the rest of our network.

The policy our NOC actually used was to believe all routes advertised by the NSFNET. Since these routes are the ones to sites attached to NSFNET regional networks, this policy was a close approximation to the far less practical to implement policy of "use NSFNET to get to those sites where the best path is through NSFNET."

The network operations center also decided to not use NSFNET at all for a few hours one afternoon when the NSFNET T-1 line was flapping. These policies generally provided good Internet communication. One major problem that we did have was that several east coast universities were unreachable at the beginning of the show. Until this problem was well understood, a number of people were blaming our routing policy for the problems. What we eventually figured out is that many of the NSFNET regional networks and many of the campuses connected to them have complex sets of routing update filters and static routes that determine how they route to various Internet networks. In at least one of the regionals, these static controls resulted in routing loops for packets destined for the show network. We also discovered fairly late in the show that this policy was resulting in suboptimal routes to BARRNet, an NSFNET regional network directly connected to NASA Ames.

Lessons

Although the network generally worked very well, there were some disappointing aspects that will, we hope, be corrected in future show networks. All stemmed from the fact that the network construction took a day longer than the schedule allowed for. For example, the schedule allowed a day for interoperability testing, but very little interoperability testing actually occurred because network construction and debugging weren't completed until shortly before the show opened. This lack of opportunity for testing discouraged vendors from doing as many inter-vendor demonstrations as I would have liked to see.

The show offered an unparalleled opportunity for using protocol analyzer to observe the behavior of a multi-vendor internet, but as I mentioned earlier the tools for this were never completely set up. The network would have been more solid had it run for a day before opening time, allowing a number of minor problems to be detected and corrected then rather than during the show.

Finally, it was disappointing to me that the vendors were in large part left to sink or swim on their own, especially since hand-holding them was what I was hired to do. However, it seemed like hand-holding was less important than having an operational network...

continued on next page

The INTEROP 88 Network *(continued)*

Despite any problems and disappointments, the network *did* work, and in fact it worked quite well. Much of the credit for this has to go to Peter de Vries, then of The Wollongong Group, who provided the hardware expertise that went into the network design and also shouldered the unenviable responsibility for building the network in five days with a crew clearly inadequate for the size of the task. Regrettably but perhaps unavoidably, TCP/IP internets are not exactly "plug and go."

Acknowledgements

Many of the best and the brightest from the Internet community, who came by during network setup to volunteer their help, made invaluable contributions. Peter and I would therefore like to recognize some of those who were particularly helpful (and apologize to those who—because we got so little sleep that week—were accidently omitted from this list):

Rick Boivie,	IBM
Len Bosack,	cisco Systems
David Bridgham,	FTP Software
Eric Brunner,	SRI International
Jeff Burgan,	The Wollongong Group
Myu Campbell,	cisco Systems
Mario Castro,	SynOptics
Shelly de Vries	
Steve Knowles,	FTP Software
Susan Hares,	Merit Computer Network
Alex Latzko,	Rutgers University
Sandy Lerner,	cisco Systems
Milo Medin,	Sterling Software
Robert Michaels,	cisco Systems
Paul Mockapetris,	USC-ISI
Mike Moesler,	Sun Microsystems
Vince Raya,	Santa Clara Convention Center
Sue Romano,	SRI International
Greg Satz,	cisco Systems
Mick Scully,	Proteon
Jim Shimoto,	Proteon
Mike St. Johns,	DCA
James VanBokkelen,	FTP Software
John Veizades,	Apple Computer

[Ed.: See also "INTEROP 88 Conference Report," in *ConneXions* Volume 2, No. 11, November 1988].

PHILIP ALMQUIST is a computer communications consultant who was hired by Advanced Computing Environments to provide vendor technical liaison and design assistance for the for the INTEROP 88 network. He also works part time at Stanford University, where he is the senior systems programmer in the campus network development group. Prior to his work on the INTEROP network, Philip played important roles in the design and construction of the Stanford University Network and of BARRNet, one of the NSFNET regional networks. His primary interest is the software infrastructure of computer networks, and he has extensive experience working on router software and the Domain Name System. He is an active member of the Internet Engineering Task Force.

Book on X Windows available

Digital Press recently announced the publication of *X Window System: C Library and Protocol Reference*, by Robert W. Scheifler, James Gettys and Ron Newman. The book was written by the people who designed and created the *X* Window System™.

The first part of the book is the reference manual for the **C** language *X* Interface Library, known as *XLib*. It presents an overview of the system, explains how to create and manipulate windows, and gives an in-depth look at the graphics capabilities. Also explained are events, event-handling functions, and a variety of utility functions.

The second part of the book is the specification of the *X* protocol semantics. It is independent from any one programming language and can be used as a starting point for creating interface libraries for other languages.

The book is intended for **C** programmers using *X*, those students looking for information on graphics, windowing systems, and user interfaces. The ISBN number is 1-55558-012-2.

Upcoming Events

Advanced Computing Environments will be hosting the *Internetworking Tutorials—TCP/IP and OSI,* April 3-6 1989 in Boston and again June 19-22 1989 in Dallas. The program is as follows:

Monday & Tuesday:

• *In-Depth Introduction to TCP/IP*	Douglas Comer
• *Berkeley UNIX Networking*	Michael Karels
• *TCP/IP for the VM Systems Programmer*	Michael Hojnowski
• *Local Area Networks and TCP/IP Alternatives*	Charles D. Brown
• *Message Handling Systems and Directory Services—X.400/X.500*	James White & Ted Myer
• *Integrated Services Digital Networks* (ISDN)*	Robert E. Blackshaw

Wednesday & Thursday:

• *The Domain Name System*	Paul Mockapetris
• *Bridges and Routers*	Radia Perlman
• *Network Management of TCP/IP-based Internets*	Jeffrey Case
• *Network Operations and Security Issues*	Eric Brunner & Ron Natalie
• *Practical Perspectives on OSI Networking*	Chris Moore & Marshall T. Rose

*Note: The ISDN tutorial will be offered *only* in April.

Mail Through the Matrix

by John S. Quarterman
Texas Internet Consulting

Introduction

There is a worldwide *metanetwork* of computer networks that use dissimilar protocols at the network or internet layer, but that communicate at the application layer. The set of such networks that are non-commercial, e.g., academic, research, or military, is sometimes called *Worldnet*. There are also some commercial networks and conferencing systems connected, and the metanetwork that includes all of these is what I call the *Matrix*. This article describes some problems associated with electronic mail correspondence through the Matrix. Although it uses RFC 822 as its main point of reference, the problems discussed are not limited to that format or to the Internet. The article is derived from material in a forthcoming book, *The Matrix: Computer Networks and Conferencing Systems Worldwide*, to be published by Digital Press.

The Service: Electronic Mail

Although the non-commercial metanetwork is closely connected, that is not because it is easy to do. Differing underlying protocols, such as those of TCP/IP, ISO-OSI, DECNET, XNS, UUCP, or SUN-III, mean that interconnection for most services would require protocol conversion, and that is not commonly done.

But there is one service that is converted and interconnected almost universally: *Electronic Mail*. This is the glue that holds the Matrix together.

Format

Mail has a simple format:

- *The body* contains the actual text of the message. The sender may prepend salutations and append closing remarks and signatures in the style of paper post, or not, as desired.Most mail systems consider the body to be straight text, although a character set or line lengths may be enforced, and Japanese systems such as JUNET distinguish several character sets within it. X.400 recognizes a hierarchical structure of data of various types. As far as the average user is concerned, mail that can be sent reliably among most systems must have a simple text body, with seven bit bytes, lines less than eighty characters long, and a single character set. The most prevalent character set is USASCII. The size of the entire message is usually limited, often to 100,000 bytes.

- *The header* contains important information such as the addresses of the sender and recipients, and a subject line, all provided by the sender, and a message identifier and date, provided by the local mail system. These are used by a local mail agent in deciding how and whether to send the mail. The subject line and date can usually be passed through most mail systems essentially unchanged. The message identifier is handled by the mail systems themselves, and should not be supplied or changed by the user. The header elements the user has to be concerned with are those containing addresses, e.g., the "From:," "To:," "Cc:," "Bcc:," "Reply-To:," and "Sender:" header lines of RFC 822.

- *The envelope* is used by a particular mail delivery system in routing and point to point delivery: this part is usually not seen by the users. Distinctions between header and envelope are often unclear, and X.400 does not appear to distinguish them at all.

Other services

Mail can be used to carry other services, because binary files can readily be encoded in hexadecimal in text that will pass through most mail services in the body of mail messages. It is quite common to transfer source files, object files, binary graphic images, and other kinds of data by this means. The encoding used must be known to the sender and recipients, but this can often be done by using a commonly-available format and noting its type in the subject header.

The Problem: Address syntax

Ideally, there would be one addressing syntax known to all networks and hosts worldwide. But the current mess is not ideal. Here is a brief summary of some of the systems in actual use.

Network addresses usually have two parts:

The *local part* specifies a mailbox for a specific user, or sometimes a distribution alias or a file to put mail in. The meaning of the local part is determined by the system specified by the host part.

The *host part* traditionally specifies a particular machine, as in `LISTSERV@BITNIC`.

There are several ways of specifying these parts, i.e., there are several commonly used separators:

`user@host` — The at sign is used in BITNET, JANET, the Internet, and many other networks. It may be the most prevalent separator.

`host::user` — This double colon syntax is used in DEC's Easynet and other networks such as MFEnet and INFnet. Note the opposite order of the local and host parts from those used with the at sign.

`host!user` — This exclamation point syntax (more commonly called *bang syntax*) is used in the UUCP network. It is unusual in more than one way, because there is often more than one element, due to the source routing used on that network.

`host1!host2!host!user` — Chains of UUCP hosts may be indicated by separating exclamation points.

`<@host2:user@host>` — This is Internet RFC 822 source routing, and can be used to accomplish the same thing as UUCP source routing: its use is much rarer, however, as is its implementation.

continued on next page

Mail Through the Matrix *(continued)*

`user%host@host2`

The percent sign here is used to do source routing very like that of RFC 822. This syntax is not required by formal mail system specifications (except JANET's *Grey Book*), but is very widely used. It depends on the general rule that the local part to the left of the at sign is interpreted locally, so that the message will reach `host2`, and that many hosts know to interpret the percent sign as a secondary at sign, so `host2` will know to send the mail on to the destination host.

Precedence

The most obvious question is: which syntax to use? The answer depends on the source network and host, the target network and host, and on any intermediate networks. Consider an address like:

```
host1!host2!hostx!user@hosta
```

This might be constructed by a user on the UUCP network to reach a user on hosta on the BITNET network. This may work, as long as all the intermediate hosts only know about UUCP bang syntax. But suppose `host2` also understands at sign syntax. The part it sees in the mail envelope will be:

```
host!user@hosta
```

Does this mean `"hostx!user"@hosta`

that is, send to hosta and expect `hosta` to do something appropriate with the local part `host!user`? If `hosta` doesn't understand bang syntax (which it won't, being on BITNET), the mail will fail.

Or does it mean `hostx!"user@hosta"`

that is, send to hostx and expect hostx to do something with the local part `user@hosta`? If `hostx` is an old-style UUCP host, it won't understand at signs and again the mail will fail.

The problem can be even worse: on TOPS-20, an exclamation point is a comment delimiter, and pairs of them are used to surround comments. Thus `host1!host2!host3!hostx!user@hosta` was read as:

```
host1host3user@hosta
```

Putting quotes around the left hand side could avoid this problem.

Even stranger things can happen when addresses with odd numbers of exclamation points are listed, as in a Cc: list:

```
hostx!userx@hosta, hosty!usery@hostb
```

would be read as

```
hostxusery@hostb
```

This could be very confusing to the sender when an error message came back from hostb for an essentially random user name. The best way around these problems is to stick to one major syntax and to convert completely at gateways. Unfortunately, many major gateways, especially ones between UUCP and the Internet, do not do this.

Some systems actually use spaces in user names. These often also accept underscore instead of space, or quotes around the whole username, but the sending user has to know this. Some systems, such as DASnet, use brackets in their addresses, while other systems, such as Telex, do not permit those characters in addresses. The user has to know appropriate substitute characters (in this case parentheses) that a gateway will transliterate.

There is, in general, no way to tell what precedence to use merely from the syntax of a mail address. And there is, in general, no way to tell what precedence a host will use. There is no way in general to tell without being told what address format can be used to reach a given user on a given system successfully. Compilations of gateways and syntaxes may be of use in getting around these problems. A generally-accepted addressing syntax is the only real solution.

Domains

A simple host part requires a global flat name space and a global host table for the network it applies to. Large, complex, or quickly changing networks can't afford this restriction. Thus domain naming systems were invented to provide hierarchical name spaces so that each part of the space could be managed by an administrative organization associated with the hosts in it.

The archetypical domain naming systems are the Internet *Domain Name System* (DNS) and JANET's *Name Registration Scheme* (NRS), which is specified in the Grey Book. They both use at signs in their syntax, for instance, `matrix@longway.tic.com` and `postmaster@uk.ac.ucl.nss`, respectively. The host part is known as the *domain part* in a domain system. The domain part is not constructed by taking a host name and appending a domain: everything to the right of the at sign is the domain. That is, it is incorrect to say that in the above example `longway` is the hostname and `tic.com` is the domain. It is true that `tic.com` is a domain, but `longway.tic.com` is also a domain, and one referring to a specific host. That is, `longway.tic.com` is the hostname in the domain naming system. That the name `longway` happens to be the old-style UUCP name for the same host is an irrelevant coincidence. If you can name a host and assume a domain to append, as in `longway` and `UUCP`, it's not a real domain.

Domains are *not* networks, despite early misconceptions on that subject and unfortunately common misuse today. A domain is an administrative entity, while a network is a technological one: a domain may include parts of many networks, and a network may include parts of many domains. The domain addresses `relay.cs.net` and `uunet.uu.net` are real: the hosts they name are the main administrative machines for CSNET and UUNET, respectively. But to name each host on CSNET `host.cs.net` would be an incorrect use of domains: the individual hosts are administered by local organizations, not by the network.

continued on next page

Mail Through the Matrix *(continued)*

Unfortunately, this distinction has not been grasped by many network administrators, and some theoretically incorrect but practically necessary examples of confusion between domains and networks do occur. The most common ones are the use of host.uucp and host.bitnet to refer to hosts on the UUCP and BITNET networks, respectively.

Order

There are three kinds of ordering problems:

- *Local and domain parts* Depending on the syntax, the local part may come first (`user@domain`) or last (`host::user`).
- *Precedence* Which of several syntactically significant operators to evaluate first may be unclear (`hostx!user@hosta`).
- *Domain order* The order of elements of a domain may be different on different networks. JANET and some other Grey Book networks use left to right order while the Internet and other DNS networks use right to left order.

Length, case, and character sets

Some networks impose very tight limits on the length of host names: old-style UUCP host names should not be longer than seven characters.

Most networks consider the host or domain part of an address to be case insensitive, i.e., `user@host` is the same as `user@HOST`. But old-style UUCP host names *are* case sensitive. The local part is sometimes case sensitive.

Europeans often use ISO8859 or other variants on ASCII to encode characters that do not occur in USASCII but that are used in their languages. These sometimes occur in local parts of network addresses. Since such characters are not alphabetic or numeric in USASCII, but instead are what are usually considered separator characters, such as the vertical bar character (|), such addresses may not be able to pass through all mail systems. Conversion into and out of different kinds of EBCDIC can also have peculiar effects, as has been noted on BITNET.

The Method: Gateways

In the current state of the networking world, one must often know a set syntax and a gateway or set of gateways to send mail through to get from one network to another. Mixing basic syntaxes should be avoided whenever possible. Mixing ones with opposite precedence, such as using ! and @ in the same address, is just asking for trouble.

Some examples:

`user%host.uucp@uunet.uu.net`
or
`user%domain@uunet.uu.net` will usually work to get mail from the Internet to the UUCP network.

`uunet!domain!user` will usually work to get mail from the UUCP network to the Internet.

`host!user@domain` will almost certainly fail to do either.

X.400 as a solution

The CCITT X.400 message handling protocol set handles a superset of all these addressing syntaxes, and might, if generally implemented, solve all these problems. But universal adoption of X.400 is not near, and meanwhile the current mess must be dealt with.

Abbreviations versus directories

Many host systems or networks allow local abbreviations of host names or whole addresses. For example, `john@here.cs.bigu.edu` might be abbreviated from hosts inside the domain `bigu.edu` as `john@here.cs` and from hosts inside `cs.bigu.edu` as `john@here` and a user who corresponded with that address frequently might establish a local abbreviation so that just `john` could be used.

But trying to use any of john, `john@here`, or `john@here.cs` from `otherhost@there.cc.stateu.edu` would almost certainly fail, or, worse, `john@here` might be interpreted as:

```
john@here.cc.stateu.edu
```

and the mail be misdelivered without warning.

Fully qualified names should always be used when trying to address anyone not extremely local (that is, on the same machine), and for location-independent uses like writing on business cards.

A Directory that allows looking up John Whoever to find an appropriate mail address is a useful idea, and a few such services exist.

The Barrier: Charging

Beyond the technical problems of interconnecting networks, there is a financial one: somebody has to pay. Costs are often hidden on the non-commercial networks, but they are explicit on the commercial ones. Gatewaying between non-commercial and commercial networks presents a special problem, in that the gateway operator has to pay for both directions. A solution used in DASnet is to allow only certain users on the non-commercial network to use the gatewaying service, and to charge them for it. But this is not a general solution to the problem, and the barrier remains, although it may be lower.

JOHN S. QUARTERMAN received an A.B. from Harvard College in 1977. He worked on networking projects for BBN from 1977 through 1980, and then for the University of Texas. Since 1986 he has been a partner in Texas Internet Consulting (TIC) of Austin, who design and install local area networks, and are involved in UNIX standards and programming. He is a member of the Board of Directors of the USENIX Association and has been involved in their networking experiments, such as UUNET.

The Navy is "on target" with OSI

by Robert Slaski, NetWorks One
and Robert Cooney, U. S. Navy

OSI across military network

The Navy has completed what is believed to be the first OSI file transfers across the Defense Data Network (DDN). The OSI FTAM Phase 1 file transfers sailed across the unclassified MILNET network between the Navy Opens Systems Laboratory at the Washington Navy Yard and the Naval Air Station in Pensacola, Florida in September 1988.

The testing was performed by the Navy Open Systems Laboratory which is staffed and managed by the Navy Regional Data Automation Center (NARDAC Washington). The OSI lab is chartered with a number of Navy OSI missions (*Communications Week*, 27 June, 1988).

The Open Systems Laboratory has been identified by the Navy as the site for the first Defense Department "application gateway" between new OSI applications and the existing TCP/IP applications. The MILNET OSI testing was the first in a series of steps aimed at outfitting the OSI application gateway. The OSI gateway provides backwards compatibility for new OSI DDN subscribers. This initial OSI gateway will permit the exchange of Navy electronic mail and data files.

DoD OSI Transition Strategy

The Office of Secretary of Defense (OSD) declared the International Organization for Standardization (ISO) Open Systems Interconnection (OSI) protocols for computer networking to be experimental co-standards with the military standard Transmission Control Protocol/Internet Protocol (TCP/IP) following the April 1987 issuance of the Government OSI Profile (GOSIP). At that time OSD directed the development of a Transition and Interoperability plan.

In the first quarter of 1988 the Defense Communication Agency completed the Department of Defense (DoD) Open Systems Interconnection Implementation Strategy. The Transition Strategy addressed the scheduling guidelines provided in the July 1987 memorandum which declared OSI as a co-standard. These scheduling guidelines are:

- Declaration of the GOSIP defined OSI profile as an experimental military co-standard effective immediately.
- Experimental interoperability between OSI and DoD protocols by March 1988.
- Limited operational interoperability between OSI and DoD protocols by January 1989.
- Mandatory procurement of OSI protocols beginning two years after the publishing of GOSIP as a Federal Information Processing Standard (August 1990).

The OSD Transition Strategy identifies an additional schedule guideline;

- Advanced OSI capabilities including X.500 Directory Services, Network Management, IS-IS routing protocols, and 1988 X.400 Message Handling System by October 1991.

The Navy demonstrated the file transfer application gateway at the *Enterprise Network Event* in Baltimore during June 1988. As a part of the OSI gateway initiative, the OSI lab staff has installed and tested a commercial electronic mail gateway between OSI X.400 and the current Simple Mail Transfer Protocol. The commercial mail gateway is a product of Sun Microsystems.

Application Gateways

The Navy also plans to provide OSI gateway services for OSI applications using the "ISO over TCP" technology supported in the ISO Development Environment (ISODE), which was pioneered by Dr. Marshall Rose of The Wollongong Group. The lab staff will field the OSI mail gateway being developed for the next release of the ISO Development Environment, (*ConneXions*, Volume 1, No. 1, May 1987). The file transfer application gateway is now publicly available as a part of ISODE version 4.0 through the University of Pennsylvania which manages ISODE distribution.

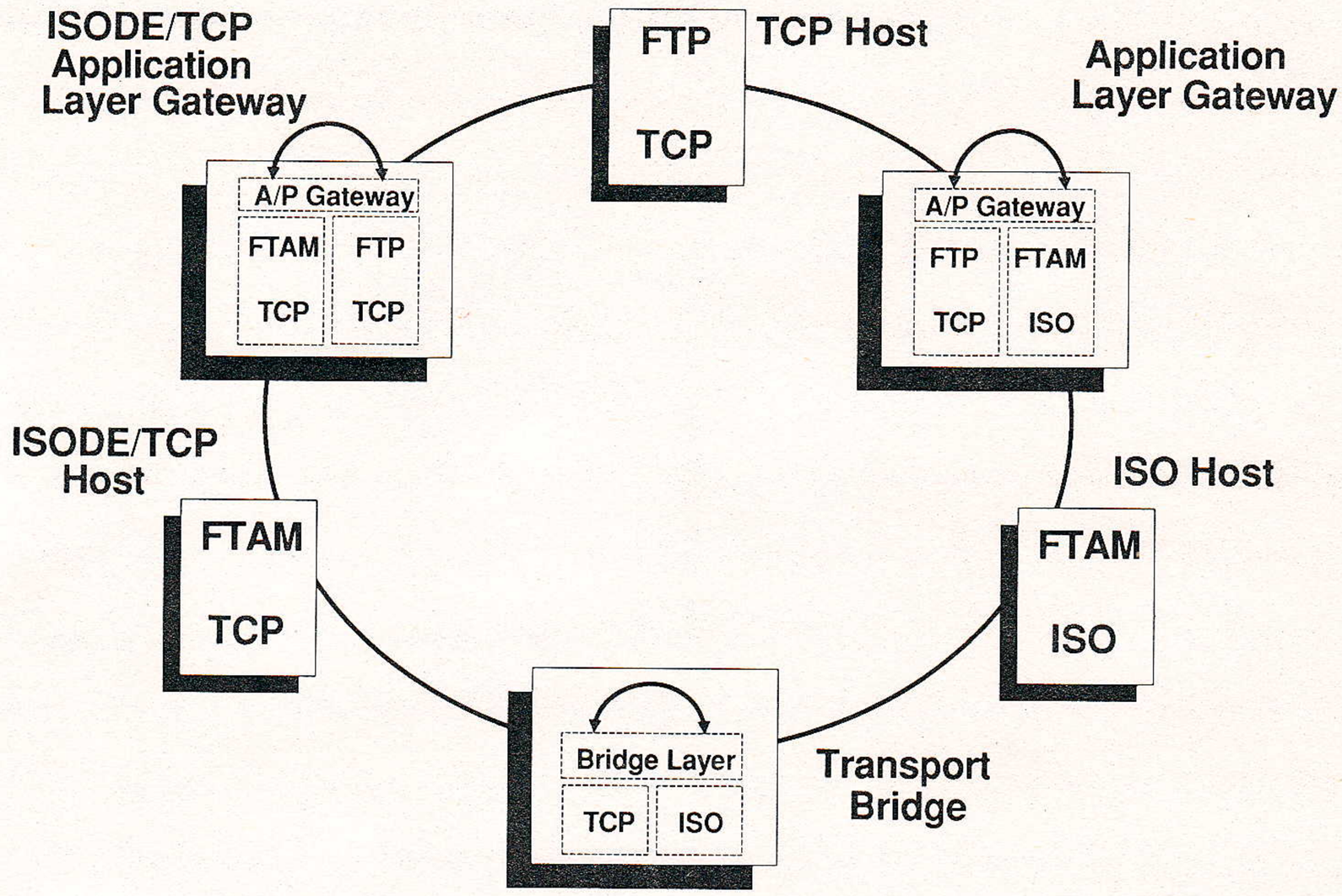

Figure 1: OSI TCP Gateway Options

The Mitre Corporation developed the file transfer application gateway in use by the Navy. The OSI laboratory is also targeted for OSI Virtual Terminal software being developed at the Mitre Corporation. The Defense Communications Agency (DCA), which is tasked with implementing the OSD Transition Strategy, is expected to follow the Navy's lead. Specifically, an application gateway similar to the Navy gateway should be operational sometime this year.

continued on next page

The Navy is "on target" with OSI *(continued)*

Why OSI for the Navy?

The Navy has been increasingly on the front line in the OSI offensive. The Navy is committed to OSI. The Navy is world wide with international bases that often use local telecommunications equipment and local support personnel. Maintenance cost will be significatly reduced if all telecommunications equipment operates according to the same standards. Use of off-the-shelf products that adhere to standards is important to the Navy as well since such products reduce both development cost and risk.

The overall benefits of OSI to the Navy are significant; less cost through competition and cost sharing with industry, faster transitioning to new standards, more efficiency, less training, and greater capabilities. The Navy has been a full participant in OSI for over 10 years.

Coexistence problems

In preparation for fielding the OSI gateway the Navy staff identified a significant incompatibility within some vendor product lines. Specifically, OSI implementations are not being designed to coexist with TCP implementations in terms of sharing a single X.25 wide area network circuit.

The Defense Data Network provides X.25 subscriber circuits which support both OSI X.25, called basic X.25, and a special military version of X.25 for TCP subscribers, called standard service. The Defense Communications Agency is expecting a single X.25 circuit to be used for both basic OSI X.25 and standard TCP X.25. Substantial delays will be incurred by military organizations which require additional circuits because combined OSI and TCP X.25 circuits are not supported by the agency network vendor.

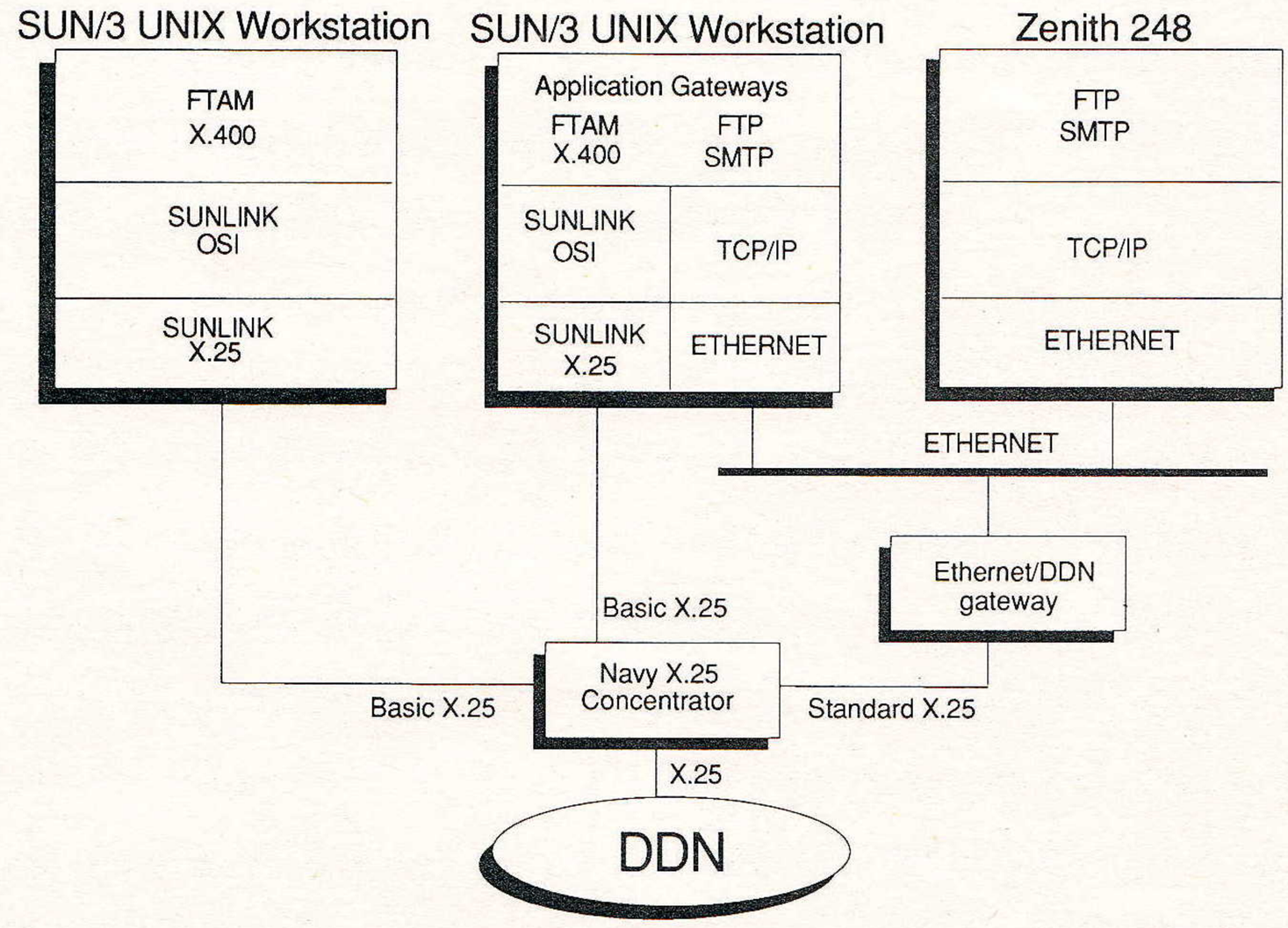

Figure 2: Navy DDN OSI Gateway configuration

This problem is addressed in the current Navy gateway by introducing an X.25 concentrator which can accept both standard and basic X.25 devices. Figure 2 shows the X.25 concentrator usage.

DDN OSI test facts

The OSI MILNET testing was performed using a Sun/3 workstation with SunLink X.25 and SunLink OSI version 5.2 and a standard military Zenith 248 microcomputer with Touch OSI version 1.0 software. The OSI profile included Transport Protocol class 4, TP4. Initial testing also included the OSI X.400, Message Handling System. In particular, X.400 electronic mail was exchanged between two workstations in the Washington area. The Navy has tested X.400 implementations by Retix and Sun Microsystems.

The Navy Open Systems Laboratory is a data communications and network test facility. Products are tested in the lab but, more importantly, the lab provides a knowledge base for Navy OSI networking efforts. The lab is supported by NetWorks One, a networking consulting company.

The ISDN Connection

The Pensacola test site is significant to the Navy since pier side Integrated Services Digital Network (ISDN) capabilities exist there which are being targeted for early OSI deployment. The overall Navy program for base telecommunications modernization is the Base Information Transfer System (BITS). Once the Navy OSI gateway is in place, communications will be possible from ships at the Pensacola pier using OSI applications to MILNET computers in Washington still using the older TCP applications.

The Navy is currently beginning a series of OSI interoperability tests with the Army and Air Force. These tests will demonstrate OSI interoperability in the actual environment required by military subscribers.

Winning the Battle

The exact content of the first DDN OSI file transfer is not known but it is rumored to have contained the single message, "Come here, Watson, I need you." It certainly seems that the OSI battle is being won.

For more information about the OSI lab contact Mr. Bob Cooney, (202) 433-5422, DDN mail to cooney@wnyosi2.arpa.

ROBERT SLASKI received a B.S. in Computer Science from the University of Florida in 1976. Mr. Slaski is currently president of NetWorks One, a networking consulting company in the Washington, D.C., area. NetWorks One is supporting the U.S. Navy Opens Systems Laboratory at the Washington Navy Yard.

ROBERT COONEY is the director of the head of the Data System Project Division at the Navy Regional Data Automation Center in Washington, D.C. Mr. Cooney manages the U.S. Navy Open Systems Laboratory which he founded. He has been actively involved in Navy OSI efforts for the last three years.

CONNEXIONS
480 San Antonio Road
Suite 100
Mountain View, CA 94040

FIRST CLASS MAIL
U.S. POSTAGE
PAID
SAN JOSE, CA
PERMIT NO. 1